A Sequential Algorithm for Solving Nonlinear Optimization Problems With Chance Constraints

Frank E. Curtis, Lehigh University

joint work with

Andreas Wächter, Northwestern University
Victor M. Zavala, University of Wisconsin–Madison

Modeling and Optimization: Theory and Applications (MOPTA)
Lehigh University, Bethlehem, Pennsylvania, USA

18 August 2016

Outline

Outline

Chance-constrained optimization

$$\min_{x \in \mathbb{R}^n} f(x)$$
$$\text{s.t. } \mathbb{P}_\xi[c(x,\xi) \leq 0] \geq 1 - \alpha$$

$f : \mathbb{R}^n \to \mathbb{R}$

$c : \mathbb{R}^n \times \Omega \to \mathbb{R}^m$

- Random variable ξ with associated space $(\Omega, \mathcal{F}, P)$
- Assume f and $c(\cdot, \xi)$ are $\mathcal{C}^1$ for any realization of ξ
- Even if f and c linear, the feasible region is nonconvex
- Assume $m = 1$ (though $m > 1$ or multiple chance constraints also OK)

Cardinality-constrained optimization

- Sample Average Approximation (SAA):

$$\Omega = \{\xi_1, \ldots, \xi_N\} \text{ with equal probability}$$

$$\begin{aligned} \min_{x \in \mathbb{R}^n} \; & f(x) \\ \text{s.t. } & |\{\xi_i \in \Omega : c(x, \xi_i) \leq 0\}| \geq \lceil (1-\alpha)N \rceil \end{aligned} \tag{CCP}$$

Cardinality-constrained optimization

- Sample Average Approximation (SAA):

$$\Omega = \{\xi_1, \ldots, \xi_N\} \text{ with equal probability}$$

$$\boxed{\begin{aligned} \min_{x \in \mathbb{R}^n} \; & f(x) \\ \text{s.t. } & |\{c_i(x) \leq 0\}| \geq M \end{aligned}} \qquad \text{(CCP)}$$

…arise in other applications as well

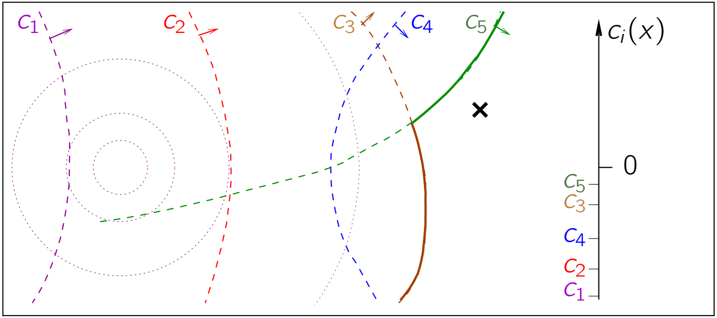
Contribution

Propose an SQP-type method for solving problem CCP

- Viable approach even when, e.g., MINLP techniques are intractable
- Like penalty-SQP, sequential minimization of penalty function
- Novel penalty function with “exactness” properties
- Subproblems with linear chance constraints
  - Do not need to “predict” optimal subset of constraints
  - Efficient methods, e.g. [Luedtke 14], [Kücükyavuz 12]
- Global convergence guarantees
- Good performance on test problems

Outline

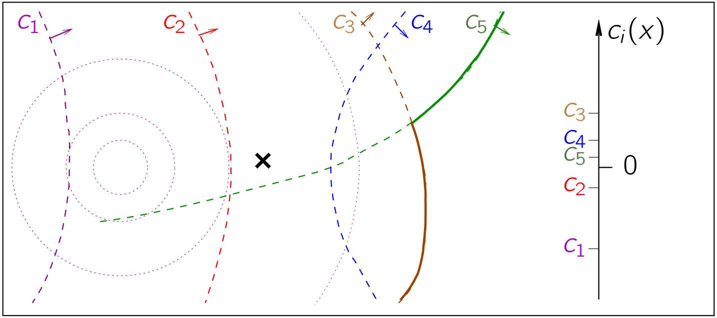
Exact penalty function for deterministic optimization



$$
\begin{aligned}
\phi(x) &= f(x) + \rho \, \| [c(x)]_+ \|_1 \\
&= f(x) + \rho \sum_{i=1}^{N} \max\{0, c_i(x)\}
\end{aligned}
$$

Under CQ, local mins of ϕ correspond to local mins of optimization problem

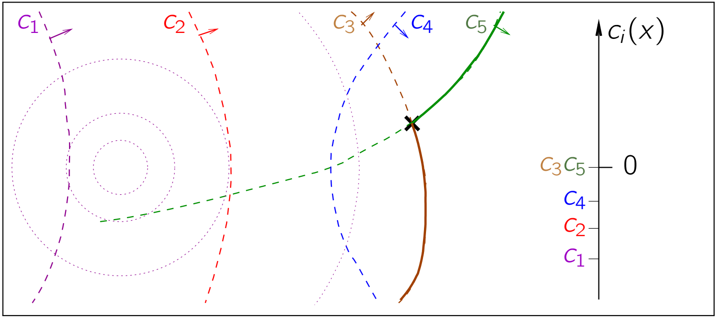
Exact penalty function for deterministic optimization



$$\phi(x) = f(x) + \rho \, \|[c(x)]_+\|_1$$
$$= f(x) + \rho \sum_{i=1}^{N} \max\{0, c_i(x)\}$$

Under CQ, local mins of ϕ correspond to local mins of optimization problem

Exact penalty function for deterministic optimization

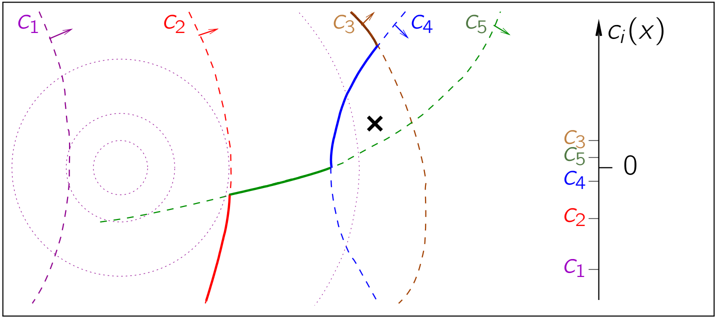


$$\phi(x) = f(x) + \rho \, \| [c(x)]_+ \|_1$$

$$= f(x) + \rho \sum_{i=1}^{N} \max\{0, c_i(x)\}$$

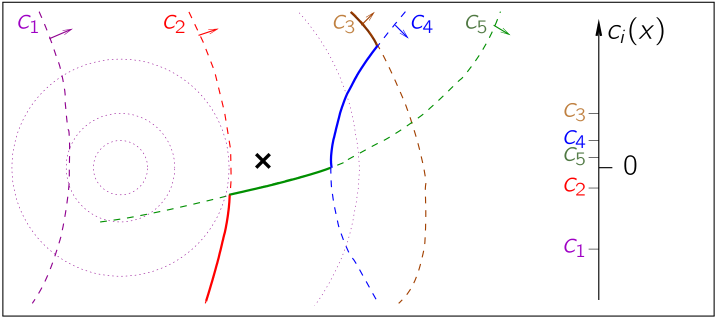
Under CQ, local mins of ϕ correspond to local mins of optimization problem

Constraint violation measure for CCP



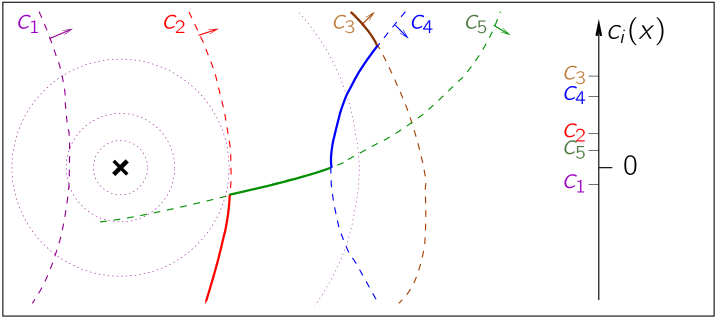
- Order constraints: $c_{i(1)}(x) \leq c_{i(2)}(x) \leq \ldots$
- Violation measure: $\langle\langle [c(x)]_+ \rangle\rangle_M = \sum_{k=1}^{M} \max\{0, c_{i(k)}(x)\}$
- Here: $\langle\langle [c(x)]_+ \rangle\rangle_M = 0$

Constraint violation measure for CCP



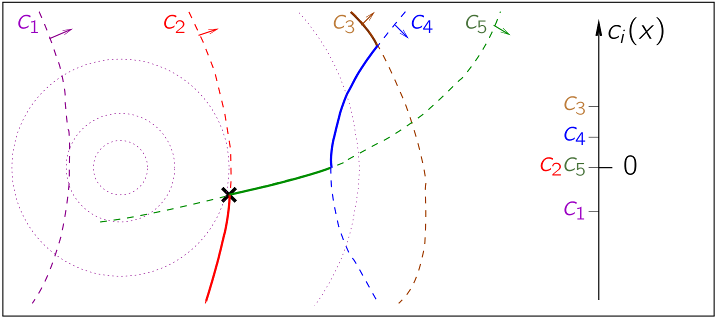
- Order constraints: $c_{i(1)}(x) \leq c_{i(2)}(x) \leq \dots$
- Violation measure: $\langle\langle [c(x)]_+ \rangle\rangle_M = \sum_{k=1}^{M} \max\{0, c_{i(k)}(x)\}$
- Here: $\langle\langle [c(x)]_+ \rangle\rangle_M = c_5(x)$

Constraint violation measure for CCP



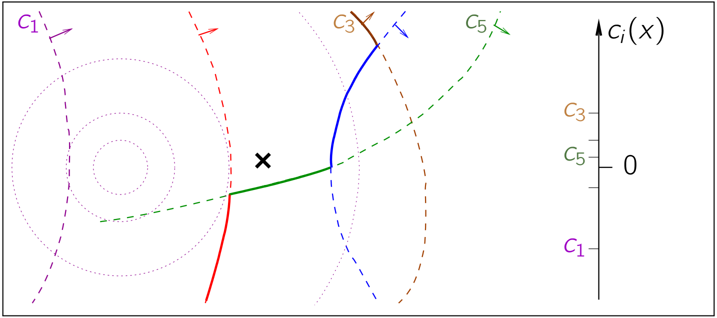
- Order constraints: $c_{i(1)}(x) \leq c_{i(2)}(x) \leq \dots$
- Violation measure: $\langle\langle [c(x)]_+ \rangle\rangle_M = \sum_{k=1}^{M} \max\{0, c_{i(k)}(x)\}$
- Here: $\langle\langle [c(x)]_+ \rangle\rangle_M = c_2(x) + c_5(x)$

Constraint violation measure for CCP



- Order constraints: $c_{i(1)}(x) \leq c_{i(2)}(x) \leq \ldots$
- Violation measure: $\langle\langle [c(x)]_+ \rangle\rangle_M = \sum_{k=1}^{M} \max\{0, c_{i(k)}(x)\}$
- Here: $\langle\langle [c(x)]_+ \rangle\rangle_M = 0$

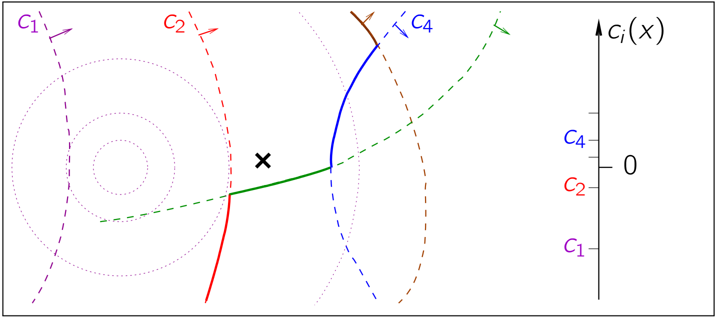
Equivalent formulation



$$\mathfrak{S} := \left\{\mathcal{S} \subseteq \{1,\ldots,N\} : |\mathcal{S}| = M\right\} \qquad c_{\mathcal{S}}(x) := [c_i(x)]_{i\in\mathcal{S}}$$

- Consider $\|[c_{\mathcal{S}}(x)]_+\|_1$ for different $\mathcal{S} \in \mathfrak{S}$
- Here: $\mathcal{S} = \{1, 3, 5\}$: $\|[c_{\mathcal{S}}(x)]_+\|_1 = c_5(x) + c_3(x)$

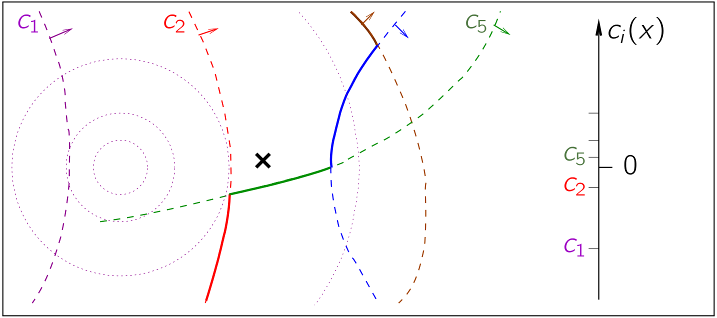
Equivalent formulation



$$\mathfrak{S} := \{\mathcal{S} \subseteq \{1,\ldots,N\} : |\mathcal{S}| = M\} \qquad c_{\mathcal{S}}(x) := [c_i(x)]_{i\in\mathcal{S}}$$

- Consider $\|[c_{\mathcal{S}}(x)]_+\|_1$ for different $\mathcal{S} \in \mathfrak{S}$
- Here: $\mathcal{S} = \{1, 2, 4\}$: $\|[c_{\mathcal{S}}(x)]_+\|_1 = c_4(x)$

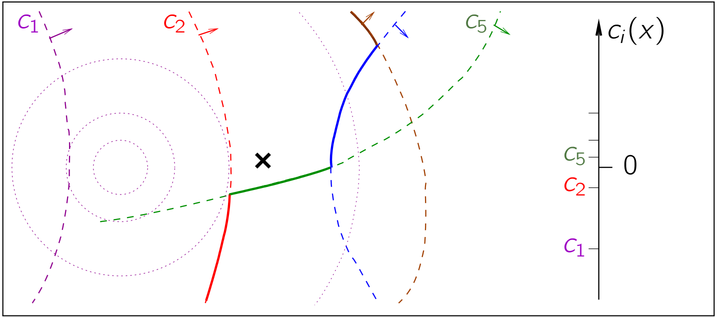
Equivalent formulation



$$\mathfrak{S} := \{\mathcal{S} \subseteq \{1, \ldots, N\} : |\mathcal{S}| = M\} \qquad c_{\mathcal{S}}(x) := [c_i(x)]_{i \in \mathcal{S}}$$

- Consider $\|[c_{\mathcal{S}}(x)]_+\|_1$ for different $\mathcal{S} \in \mathfrak{S}$
- Here: $\mathcal{S} = \{1, 2, 5\}$: $\|[c_{\mathcal{S}}(x)]_+\|_1 = c_5(x)$

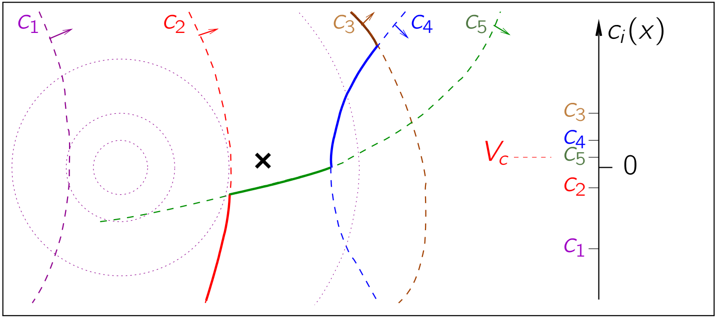
Equivalent formulation



$$\mathfrak{S} := \left\{\mathcal{S} \subseteq \{1, \ldots, N\} : |\mathcal{S}| = M\right\} \qquad c_{\mathcal{S}}(x) := [c_i(x)]_{i \in \mathcal{S}}$$

- Consider $\|[c_{\mathcal{S}}(x)]_+\|_1$ for different $\mathcal{S} \in \mathfrak{S}$
- So $\langle\!\langle [c(x)]_+ \rangle\!\rangle_M = \min_{\mathcal{S} \in \mathfrak{S}} \|[c_{\mathcal{S}}(x)]_+\|_1$

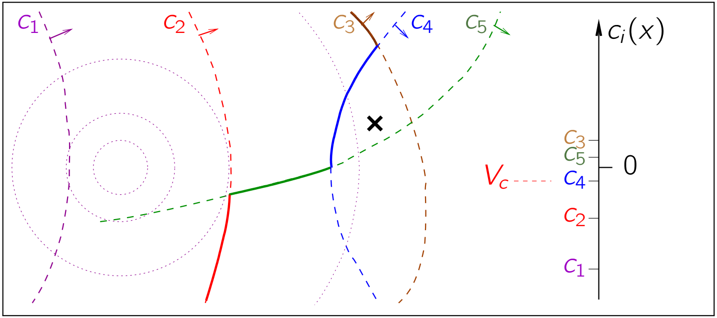
Critical scenario selections



$$\mathfrak{S}(x) = \{\mathcal{S} \in \mathfrak{S} : c_i(x) \leq V_c(x) \text{ for } i \in \mathcal{S}\} \qquad V_c(x) = c_{i(M)}(x)$$

- Then $\langle\langle [c(x)]_+ \rangle\rangle_M = \|[c_{\mathcal{S}}(x)]_+\|_1$ for all $\mathcal{S} \in \mathfrak{S}(x)$.
- Here: $\mathfrak{S}(x) = \{(1, 2, 5)\}$

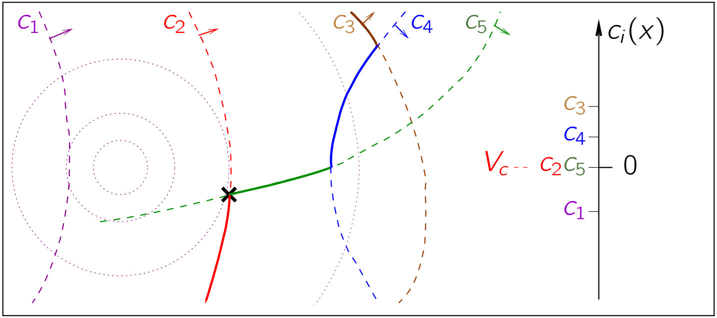
Critical scenario selections



$$\mathfrak{S}(x) = \{\mathcal{S} \in \mathfrak{S} : c_i(x) \leq V_c(x) \text{ for } i \in \mathcal{S}\} \qquad V_c(x) = c_{i(M)}(x)$$

- Then $\langle\!\langle [c(x)]_+ \rangle\!\rangle_M = \|[c_{\mathcal{S}}(x)]_+\|_1$ for all $\mathcal{S} \in \mathfrak{S}(x)$.
- Here: $\mathfrak{S}(x) = \{(1, 2, 4)\}$

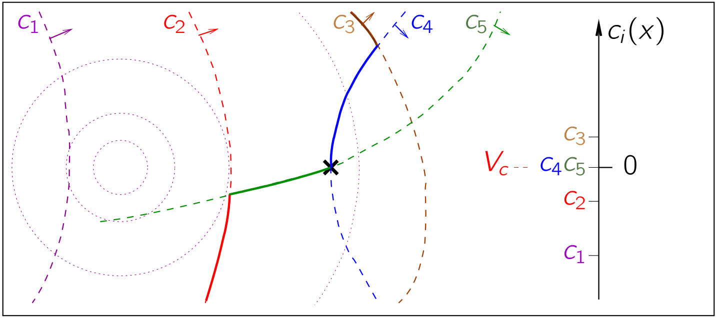
Critical scenario selections



$$\mathfrak{S}(x) = \{\mathcal{S} \in \mathfrak{S} : c_i(x) \leq V_c(x) \text{ for } i \in \mathcal{S}\} \qquad V_c(x) = c_{i(M)}(x)$$

- Then $\langle\langle [c(x)]_+ \rangle\rangle_M = \|[c_{\mathcal{S}}(x)]_+\|_1$ for all $\mathcal{S} \in \mathfrak{S}(x)$.
- Here: $\mathfrak{S}(x) = \{(1, 2, 5)\}$

Critical scenario selections



$$\mathfrak{S}(x) = \{\mathcal{S} \in \mathfrak{S} : c_i(x) \leq V_c(x) \text{ for } i \in \mathcal{S}\} \qquad V_c(x) = c_{i(M)}(x)$$

- Then $\langle\langle [c(x)]_+ \rangle\rangle_M = \|[c_{\mathcal{S}}(x)]_+\|_1$ for all $\mathcal{S} \in \mathfrak{S}(x)$.
- Here: $\mathfrak{S}(x) = \{(1, 2, 4), (1, 2, 5)\}$

Exact penalty function for CCP

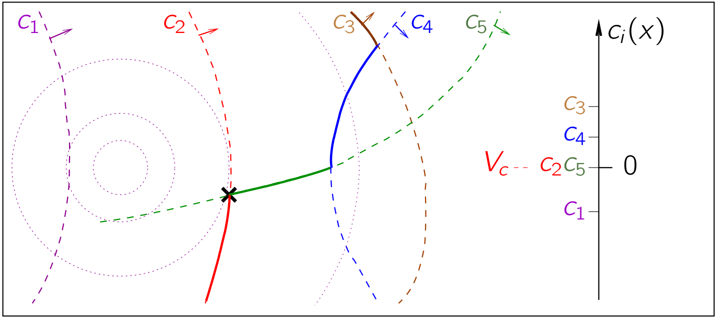
Penalty function for deterministic optimization with constraints in $\mathcal{S} \in \mathfrak{S}$:

$$\phi_{\mathcal{S}}(x) := f(x) + \rho \|[c_{\mathcal{S}}(x)]_+\|_1$$

Penalty function for CCP:

$$\begin{aligned}
\phi(x) &= f(x) + \rho \, \langle\!\langle [c(x)]_+ \rangle\!\rangle_M \\
&= f(x) + \rho \min_{\mathcal{S} \in \mathfrak{S}} \; \|[c_{\mathcal{S}}(x)]_+\|_1 \\
&= \min_{\mathcal{S} \in \mathfrak{S}} \; f(x) + \rho \|[c_{\mathcal{S}}(x)]_+\|_1 \\
&= \min_{\mathcal{S} \in \mathfrak{S}} \; \phi_{\mathcal{S}}(x) \\
&= \phi_{\tilde{\mathcal{S}}}(x) \qquad\qquad (\text{for all } \tilde{\mathcal{S}} \in \mathfrak{S}(x))
\end{aligned}$$

Equivalence of minimizers

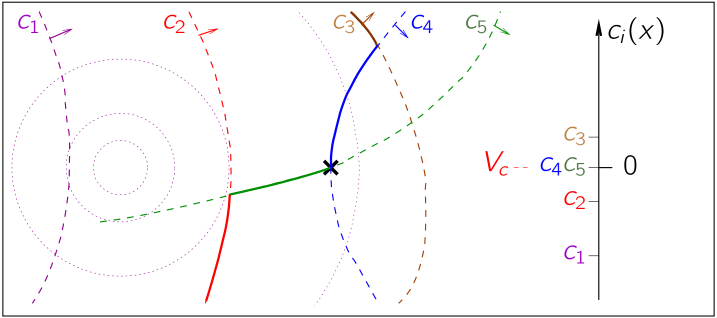


For $\rho > 0$ sufficiently large (and some CQ):

$$x_* \text{ local min of CCP}$$
$$\Updownarrow$$
$$\langle\langle [c(x_*)]_+ \rangle\rangle_M = 0 \text{ and } x_* \text{ local min of } \phi_{\mathcal{S}}(x) \text{ for all } \mathcal{S} \in \mathfrak{S}(x_*)$$

- Here $\mathfrak{S}(x_*) = \{(1, 2, 5)\}$ and x_* is local min of CCP

Equivalence of minimizers



For $\rho > 0$ sufficiently large (and some CQ):

$$x_* \text{ local min of CCP}$$
$$\Updownarrow$$
$$\langle\langle [c(x_*)]_+ \rangle\rangle_M = 0 \text{ and } x_* \text{ local min of } \phi_{\mathcal{S}}(x) \text{ for all } \mathcal{S} \in \mathfrak{S}(x_*)$$

- Here $\mathfrak{S}(x_*) = \{(1, 2, 4), (1, 2, 5)\}$ and x_* is NOT local min of CCP

Outline

Trust-region method for deterministic optimization (Sℓ_1QP)

$$\phi(x) = f(x) + \rho\|[c(x)]_+\|_1 \qquad (g_k = \nabla f(x_k), J_k = \nabla c(x_k)^T)$$

$$m_k(d) = f_k + g_k^T d + \tfrac{1}{2} d^T H_k d + \rho\|[c_k + J_k d]_+\|_1$$

At iterate x_k:

1. Compute step d_k from trust region subproblem (QP):

$$\boxed{\min_{d \in \mathbb{R}^n} \; m_k(d) \quad \text{s.t.} \quad \|d\|_\infty \le \delta_k}$$

2. Define ratio

$$r_k = \frac{\phi(x_k) - \phi(x_k + d_k)}{m_k(0) - m_k(d_k)}$$

3. Update iterate and trust region radius $(\mu \in (0,1))$
   - If $r_k \ge \mu$: $x_{k+1} \leftarrow x_k + d_k$ and $\delta_{k+1} \leftarrow 2\delta_k$
   - If $r_k < \mu$: $x_{k+1} \leftarrow x_k$ and $\delta_{k+1} \leftarrow 0.5\|d_k\|_\infty$

Trust-region method for CCP

$$\phi(x) = f(x) + \rho\langle\langle [c(x)]_+ \rangle\rangle_M$$
$$m_k(d) = f_k + g_k^T d + \tfrac{1}{2} d^T H_k d + \rho\langle\langle [c_k + J_k d]_+ \rangle\rangle_M$$

At iterate x_k:

1. Compute step d_k from trust region subproblem (QCCP):

$$\min_{d\in\mathbb{R}^n} \ m_k(d) \quad \text{s.t.} \quad \|d\|_\infty \le \delta_k$$

2. Define ratio

$$r_k = \frac{\phi(x_k) - \phi(x_k + d_k)}{m_k(0) - m_k(d_k)}$$

3. Update iterate and trust region radius $(\mu \in (0,1))$
   - If $r_k \ge \mu$: $x_{k+1} \leftarrow x_k + d_k$ and $\delta_{k+1} \leftarrow \max\{2\delta_k, \delta_{\text{reset}}\}$ $(\delta_{\text{reset}} > 0)$
   - If $r_k < \mu$: $x_{k+1} \leftarrow x_k$ and $\delta_{k+1} = 0.5\|d_k\|_\infty$

Solving the trust region subproblem

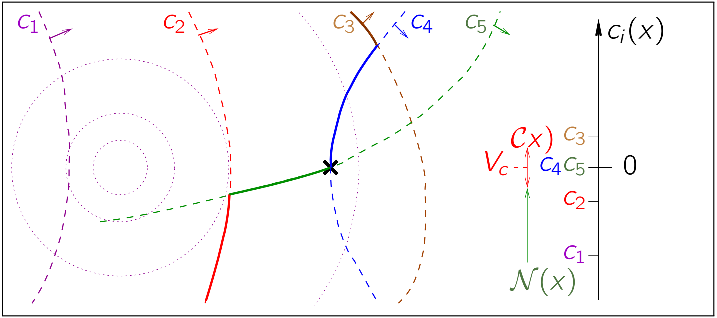
$$m_k(d) = f_k + g_k^T d + \tfrac{1}{2} d^T H_k d + \rho \, \langle\!\langle [c_k + J_k d]_+ \rangle\!\rangle_M$$
$$= f_k + g_k^T d + \tfrac{1}{2} d^T H_k d + \rho \min_{\mathcal{S} \in \mathfrak{S}} \| [c_{\mathcal{S},k} + J_{\mathcal{S},k} d]_+ \|_1$$

MIQP formulation:

$$\min_{d,s,z} \; f_k + g_k^T d + \tfrac{1}{2} d^T H_k d + \rho \sum_{i=1}^{N} s_i$$
$$\text{s.t. } c_i(x_k) + \nabla c_i(x_k)^T d \le s_i + \text{bigM}(1 - z_i)$$
$$\sum_{i=1}^{N} z_i = M, \quad s \ge 0, \quad \|d\|_\infty \le \delta_k, \quad z \in \{0,1\}^N$$

- big-M formulation is computationally expensive
- Better alternative: Use B&C method of [Luedtke 14]
- Can we reduce the number of scenarios to consider?

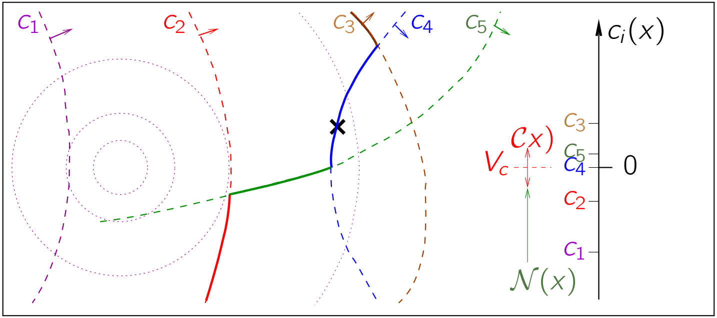
ϵ-critical scenario selections



$$\mathcal{C}(x,\epsilon) = \{i \in \{1,\ldots,N\} : |c_i(x) - V_c(x)| \le \epsilon\}$$
$$\mathcal{N}(x,\epsilon) = \{i \in \{1,\ldots,N\} : c_i(x) \le V_c(x) - \epsilon\}$$
$$\mathfrak{S}(x,\epsilon) = \{\mathcal{S} \in \mathfrak{S} : \mathcal{N}(x,\epsilon) \subseteq \mathcal{S} \text{ and } \mathcal{S} \subseteq \mathcal{N}(x,\epsilon) \cup \mathcal{C}(x,\epsilon)\}$$

- Here: $\mathfrak{S}(x_*,\epsilon) = \{(1,2,4),(1,2,5)\}$

ϵ-critical scenario selections



$$\mathcal{C}(x,\epsilon) = \{i \in \{1,\ldots,N\} : |c_i(x) - V_c(x)| \le \epsilon\}$$
$$\mathcal{N}(x,\epsilon) = \{i \in \{1,\ldots,N\} : c_i(x) \le V_c(x) - \epsilon\}$$
$$\mathfrak{S}(x,\epsilon) = \{\mathcal{S} \in \mathfrak{S} : \mathcal{N}(x,\epsilon) \subseteq \mathcal{S} \text{ and } \mathcal{S} \subseteq \mathcal{N}(x,\epsilon) \cup \mathcal{C}(x,\epsilon)\}$$

- Here: $\mathfrak{S}(x_k,\epsilon) = \{(1,2,4),(1,2,5)\}$

Subproblem with ϵ-critical scenario selections

$$\min_{d \in \mathbb{R}^n} \; f_k + g_k^T d + \tfrac{1}{2} d^T H_k d + \rho \min_{\mathcal{S} \in \mathfrak{S}(x_k, \epsilon)} \| [c_{\mathcal{S},k} + J_{\mathcal{S},k} d]_+ \|_1$$

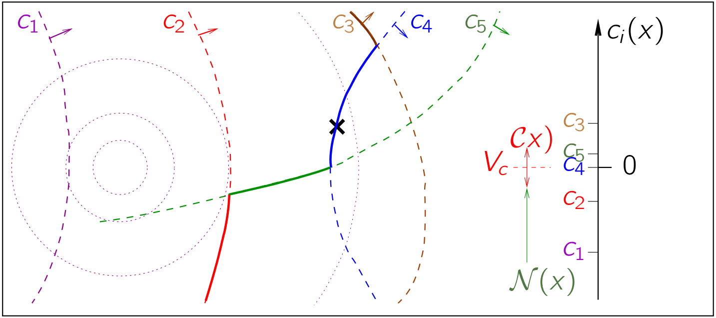
$$\text{s.t. } \|d\|_\infty \le \delta_k$$

- Computational effort can be tuned with ϵ
  - Small ϵ: Easier subproblems
  - Large ϵ: "Broader view" of feasible region

Theorem 1

Suppose $f \in \mathcal{C}^1$, $c(\cdot, \xi) \in \mathcal{C}^1$, $\{H_k\}$ bounded, and $\phi(x_k)$ bounded below. Then, every limit point x_* of $\{x_k\}$ is a stationary point of $\phi_{\mathcal{S}}$ for all $\mathcal{S} \in \mathfrak{S}(x_*)$.

Joint chance constraints



- Joint chance constraints: $m > 1$
- Combined constraint satisfaction:

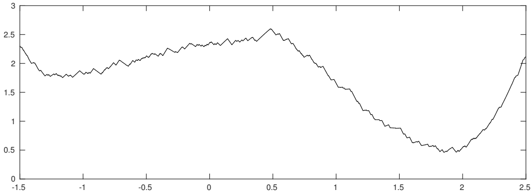
$$V_i(x) = \begin{cases} \|[c_i(x)]_+\|_1 & \text{if } c_i(x) \not\leq 0 \\ \max\{c_{ij}(x) : j = 1, \ldots, m\} & \text{if } c_i(x) \leq 0 \end{cases}$$

Outline

Numerical experiments

- Matlab implementation with AMPL interface (suffixes)
- MIQPs solved by CPLEX
  - take incumbent after 5 min time limit (parallel with 8 threads)
- H_k: Hessian of Lagrangian for current scenario selection
- $\gamma \in (0, 1]$: include $\lceil \gamma N \rceil$ scenarios below and above $V_c(x_k)$ in $\mathcal{C}$

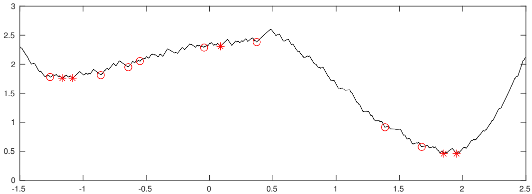
"Poor" local minimizers



$$\min_{x,z} \; z$$
$$\text{s.t. } \mathbb{P}[0.25x^4 - \tfrac{1}{3}x^3 - x^2 + 0.2x - 19.5 + \xi_1 x + \xi_1 \xi_0 \leq z] \geq 0.95$$

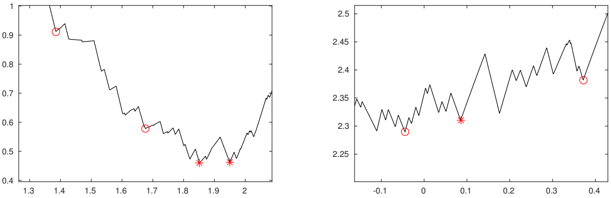
- ξ_0 from $U[-12, 12]$ and ξ_1 from $U[-3, 3]$
- $N = 5{,}000$

“Poor” local minimizers



γ		0.001	0.002	0.005	0.010	0.050	0.100	0.200	1.000
x_0	N	11	21	51	101	501	751	1251	5000
−1.5		*1.776*	*1.776*	**1.754**	**1.754**	**1.754**	**1.754**	**1.754**	**1.754**
−1.0		**1.754**	**1.760**	**1.754**	**1.754**	**1.754**	**1.754**	**1.754**	**1.754**
−0.5		2.049	1.951	1.814	**1.754**	**1.754**	**1.754**	**1.754**	**1.754**
0.0		2.289	2.289	1.814	**1.754**	**1.754**	**1.754**	**1.754**	**1.754**
0.5		2.382	2.382	**0.460**	**0.460**	**0.460**	**0.460**	**0.460**	**0.460**
1.0		0.912	**0.460**	**0.460**	**0.460**	**0.463**	**0.463**	**0.463**	**0.463**
1.5		0.579	**0.460**	**0.460**	**0.460**	**0.460**	**0.460**	**0.460**	**0.460**
2.0		**0.463**	**0.463**	**0.460**	**0.460**	**0.463**	**0.463**	**0.463**	**0.463**
x^{bad}		2.310	2.289	1.814	**1.754**	**1.754**	**1.754**	**1.754**	**1.754**

“Poor” local minimizers



x_0 \ γ / N	0.001 / 11	0.002 / 21	0.005 / 51	0.010 / 101	0.050 / 501	0.100 / 751	0.200 / 1251	1.000 / 5000
-1.5	*1.776*	*1.776*	**1.754**	**1.754**	**1.754**	**1.754**	**1.754**	**1.754**
-1.0	**1.754**	**1.760**	**1.754**	**1.754**	**1.754**	**1.754**	**1.754**	**1.754**
-0.5	2.049	1.951	1.814	**1.754**	**1.754**	**1.754**	**1.754**	**1.754**
0.0	2.289	2.289	1.814	**1.754**	**1.754**	**1.754**	**1.754**	**1.754**
0.5	2.382	2.382	**0.460**	**0.460**	**0.460**	**0.460**	**0.460**	**0.460**
1.0	0.912	**0.460**	**0.460**	**0.460**	**0.463**	**0.463**	**0.463**	**0.463**
1.5	0.579	**0.460**	**0.460**	**0.460**	**0.460**	**0.460**	**0.460**	**0.460**
2.0	**0.463**	**0.463**	**0.460**	**0.460**	**0.463**	**0.463**	**0.463**	**0.463**
x^{bad}	2.310	2.289	1.814	**1.754**	**1.754**	**1.754**	**1.754**	**1.754**

Cash flow problem [Dentcheva et al. 03]

- Invest capital over time horizon into different investment options
- Make sure we can pay random liabilities (with 95% prob.)
- Maximize final cash
- Interest rates depend on amount of investment
  - This makes the problem nonconvex
- 5 random instances per size

Results

N	γ	instances solved	iter	$\|\mathcal{C}_k\|$	changes in $\mathcal{C}_k$	% improve over robust	time in secs
500	0.001	5	9.84	9.40	6.14	5.7223 (5)	25.95
500	0.005	5	8.19	11.84	8.68	5.9602 (5)	37.45
500	0.050	5	7.04	51.00	13.06	6.0349 (5)	52.96
500	1.000	5	7.04	500.00	15.52	6.0349 (5)	260.08
1000	0.001	5	15.98	10.90	15.15	7.4040 (5)	117.13
1000	0.005	5	12.29	15.22	23.72	7.7296 (5)	147.30
1000	0.050	5	8.12	101.00	34.29	7.9134 (5)	313.17
1000	1.000	5	7.43	1000.00	57.14	7.9134 (5)	819.24
2000	0.001	5	21.94	13.84	43.67	9.7347 (4)	652.19
2000	0.005	4	12.85	23.01	56.61	9.9029 (4)	513.20
2000	0.050	4	9.90	201.00	88.99	10.1009 (4)	1099.41
2000	1.000	5	14.33	2000.00	129.18	10.0924 (4)	3519.53

- 96 vars, 20 chance constr, 150 non-chance constr
- Size for $N = 2000$ and $\gamma = 0.20$:
  - MIQP has about 40,000 constraints and 500 binary variables

Outline

Conclusions

- SQP-type trust-region algorithm
  - Exact penalty function for chance-constrained NLP
- Computational effort of subproblem can be tuned
  - Small ϵ: Reduce solution time
  - Large ϵ: Escape spurious local solutions
- Potential improvements
  - Use efficient branch-and-cut subproblem (e.g., [Luedtke 12], ...)
  - Adaptive choices of ρ and/or ϵ or γ
  - Adaptive sample size
    - At beginning, consider small subset of scenarios and choose large ϵ
    - Refine SAA discretization with proximity to solution
  - Heuristic solutions of subproblem
    - Avoid solution of mixed-integer problem
    - E.g., progress along successively improving scenario selections
  - Ignore "very satisfied" constraints in subproblem
    - Do not include scenarios with $c_i(x) \ll 0$